IBM Developer
SKILLS NETWORK

Winning Space Race with Data Science

Kejie Huang
Feb 2023

Outline

- Executive Summary
- Introduction
- Methodology
- Results
- Conclusion
- Appendix

Executive Summary

- Summary of methodologies
  - Data Collection through API
  - Data Collection with Web Scraping
  - Data Wrangling
  - Exploratory Data Analysis with SQL
  - Exploratory Data Analysis with Data Visualization
  - Interactive Visual Analytics with Folium
  - Machine Learning Prediction
- Summary of all results
  - EDA result
  - Interactive Visual Analytics with Geological Information analytics
  - Predictive Analytics Result with Machine Learning
  - Python Data Dashboard App

Introduction

- Project background and context

  SpaceX advertises Falcon 9 rocket launches on its website with a cost of 62 million dollars; other providers cost upward of 165 million dollars each, much of the savings is because SpaceX can reuse the first stage. Therefore, if we can determine if the first stage will land, we can determine the cost of a launch. This information can be used if an alternate company wants to bid against SpaceX for a rocket launch. In this lab, you will collect and make sure the data is in the correct format from an API. This goal of the project is to create a data science process to predict if the Falcon 9 first stage will land successfully.

- Problems you want to find answers

  - Estimate the total cost of new rocket launches.
  - What factors affect the cost and what is the trade-off between them?
  - In what case will first stage be sacrificed.

Section 1

Methodology

Methodology

Executive Summary

- Data collection methodology.
  - Web data scraping using python.
- Perform data wrangling
- Perform exploratory data analysis (EDA) using visualization and SQL
  - Create a database for data storage and easiness of access
- Perform interactive visual analytics using Folium and Plotly Dash App
- Perform predictive analysis using classification models
  - Apply common types of learning models and find solution with the best one.

Data Collection

- The data was collected using various methods
  - Data collection was done through the SpaceX API and web pages of Wikipedia.
  - Decode the server content in Json and transfer into a pandas dataframe.
  - Applied web scraping from Wikipedia for Falcon 9 launch records, clean and summarized data.

Data Collection – SpaceX API

- Used SpaceX API to gather data, done basic data wrangling and formatting to make data useful for next step.
- Ipynb link.
- [CapstoneDataScience/Complete the Data Collection API Lab.ipynb at master · mg4234/CapstoneDataScience (github.com)]()

Out[26]:

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Lc
4	1	2010-06-04	Falcon 9	NaN	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0003	-8
5	2	2012-05-22	Falcon 9	525.0	LEO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0005	-8
6	3	2013-03-01	Falcon 9	677.0	ISS	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B0007	-8
7	4	2013-09-29	Falcon 9	500.0	PO	VAFB SLC 4E	False Ocean	1	False	False	False	None	1.0	0	B1003	-12
8	5	2013-12-03	Falcon 9	3170.0	GTO	CCSFS SLC 40	None None	1	False	False	False	None	1.0	0	B1004	-8
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
89	86	2020-09-03	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	2	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	7	B1060	-8
90	87	2020-10-06	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	3	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	7	B1058	-8
91	88	2020-10-18	Falcon 9	15600.0	VLEO	KSC LC 39A	True ASDS	6	True	True	True	5e9e3032383ecb6bb234e7ca	5.0	9	B1051	-8
92	89	2020-10-24	Falcon 9	15600.0	VLEO	CCSFS SLC 40	True ASDS	3	True	True	True	5e9e3033383ecbb9e534e7cc	5.0	7	B1060	-8
93	90	2020-11-05	Falcon 9	3681.0	MEO	CCSFS SLC 40	True ASDS	1	True	False	True	5e9e3032383ecb6bb234e7ca	5.0	2	B1062	-8

90 rows × 17 columns

Data Wrangling

We can see below that some of the rows are missing values in our dataset.

In [27]:

```
data_falcon9.isnull().sum()
```

Out[27]:

```
FlightNumber      0
Date              0
BoosterVersion    0
PayloadMass       5
Orbit             0
LaunchSite        0
Outcome           0
Flights           0
GridFins          0
Reused            0
Legs              0
LandingPad       26
Block             0
ReusedCount       0
Serial            0
Longitude         0
Latitude          0
```

Data Collection - Scraping

- Collected SpaceX launching data from Wikipedia webpage
- Parsed tables on web page and converted from HTML it into pandas dataframes.
- Ipynb link:
- [CapstoneDataScience/Data Collection with Web Scraping lab.ipynb at master · mg4234/CapstoneDataScience (github.com)]

In [22]:

```
headings = []
for key,values in dict(launch_dict).items():
    if key not in headings:
        headings.append(key)
    if values is None:
        del launch_dict[key]

def pad_dict_list(dict_list, padel):
    lmax = 0
    for lname in dict_list.keys():
        lmax = max(lmax, len(dict_list[lname]))
    for lname in dict_list.keys():
        ll = len(dict_list[lname])
        if  ll < lmax:
            dict_list[lname] += [padel] * (lmax - ll)
    return dict_list

pad_dict_list(launch_dict,0)

df = pd.DataFrame(launch_dict)
df.head()
```

Out[22]:

	Flight No.	Launch site	Payload	Payload mass	Orbit	Customer	Launch outcome	Version Booster	Booster landing	Date	Time
0	1	CCAFS	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success\n	F9 v1.0B0003.1	Failure	4 June 2010	18:45
1	2	CCAFS	Dragon	0	LEO	NASA	Success	F9 v1.0B0004.1	Failure	8 December 2010	15:43
2	3	CCAFS	Dragon	525 kg	LEO	NASA	Success	F9 v1.0B0005.1	No attempt\n	22 May 2012	07:44
3	4	CCAFS	SpaceX CRS-1	4,700 kg	LEO	NASA	Success\n	F9 v1.0B0006.1	No attempt	8 October 2012	00:35
4	5	CCAFS	SpaceX CRS-2	4,877 kg	LEO	NASA	Success\n	F9 v1.0B0007.1	No attempt\n	1 March 2013	15:10

We can now export it to a **CSV** for the next section, but to make the answers consistent and in case you have difficulties finishing this lab.

Following labs will be using a provided dataset to make each lab independent.

In [23]:

```
df.to_csv('spacex_web_scraped.csv', index=False)
```



```
df.to_csv('spacex_web_scraped.csv', index=False)
```

Data Wrangling

- Performed basic data process to help finding the training labels for next steps.
- Calculated the number of launches at each site.
- Number and occurrence of each orbits
- We created landing outcome label from outcome column
- Ipynb link:
- [CapstoneDataScience/EDA lab.ipynb at master · mg4234/CapstoneDataScience (github.com)]

This variable will represent the classification variable that represents the outcome of each launch. If the value is zero, the first stage did not land successfully; one means the first stage landed Successfully

```
df['Class']=landing_class
df[['Class']].head(8)
```

	Class
0	0
1	0
2	0
3	0
4	0
5	0
6	1
7	1

```
df.head(5)
```

	FlightNumber	Date	BoosterVersion	PayloadMass	Orbit	LaunchSite	Outcome	Flights	GridFins	Reused	Legs	LandingPad	Block	ReusedCount	Serial	Longitude	Latit
0	1	2010-06-04	Falcon 9	6104.959412	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0003	-80.577366	28.561
1	2	2012-05-22	Falcon 9	525.000000	LEO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0005	-80.577366	28.561
2	3	2013-03-01	Falcon 9	677.000000	ISS	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B0007	-80.577366	28.561
3	4	2013-09-29	Falcon 9	500.000000	PO	VAFB SLC 4E	False Ocean	1	False	False	False	NaN	1.0	0	B1003	-120.610829	34.632
4	5	2013-12-03	Falcon 9	3170.000000	GTO	CCAFS SLC 40	None None	1	False	False	False	NaN	1.0	0	B1004	-80.577366	28.561

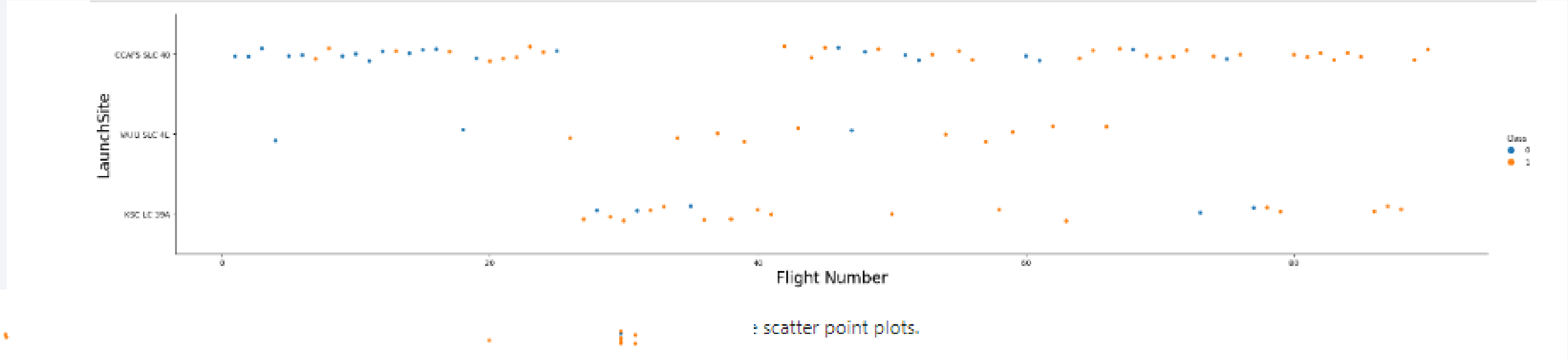
We can use the following line of code to determine the success rate:

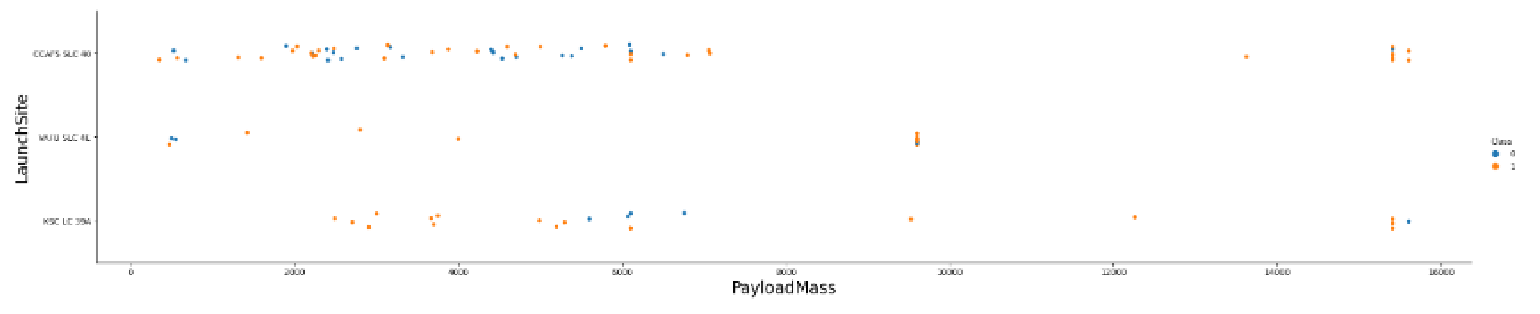
```
df["Class"].mean()
```

0.6666666666666666

EDA with Data Visualization

- Used pandas and matplotlib to create scatter and bar charts to We explored the data by visualizing the relationship between factors.





- Ipynb link:
  - [CapstoneDataScience/EDA with Visualization lab.ipynb at master · mg4234/CapstoneDataScience (github.com)]

EDA with SQL

- Setup a database on IBM cloud to store the SpaceX dataset for access.
- Used Python SQL API to query data with EDA method to get insights of data.
- Some of queries:
    - **names of the unique launch sites in the space mission**
    - **launch sites begin with the string 'CCA'**
    - **total payload mass carried by boosters launched by NASA (CRS)**
    - **average payload mass carried by booster version F9 v1.1**
    - **first successful landing outcome in ground pad was acheived**
    - **names of the boosters which have success in drone ship and have payload mass greater than 4000 but less than 6000**
- **Ipynb link**
- CapstoneDataScience/EDA with SQL lab.ipynb at master · mg4234/CapstoneDataScience (github.com)

Build an Interactive Map with Folium

- Used Folium Map app to mark launch sites on an interactive map.
- Marked the success or failure of launches in groups for each site on map.
- We calculated the distances between a launch site to proximities:
    - Launch site to coastline
    - Launch site to nearest railway
- Ipynb link:
- [CapstoneDataScience/Interactive Visual Analytics with Folium lab.ipynb at master · mg4234/CapstoneDataScience (github.com)]

Build a Dashboard with Plotly Dash

- Create an interactive dashboard with Plotly dash
- Allow user to plotted pie charts and scatter plot in this GUI like page.
- Ipynb link:
- [CapstoneDataScience/spacex_dash_app.py at master · mg4234/CapstoneDataScience (github.com)]

Predictive Analysis (Classification)

- Used 4 classification models/methods to test the data set and find the one with best accuracy by comparison.
- Ipynb link:
- CapstoneDataScience/Machine Learning Prediction lab.ipynb at master · mg4234/CapstoneDataScience (github.com)

Results

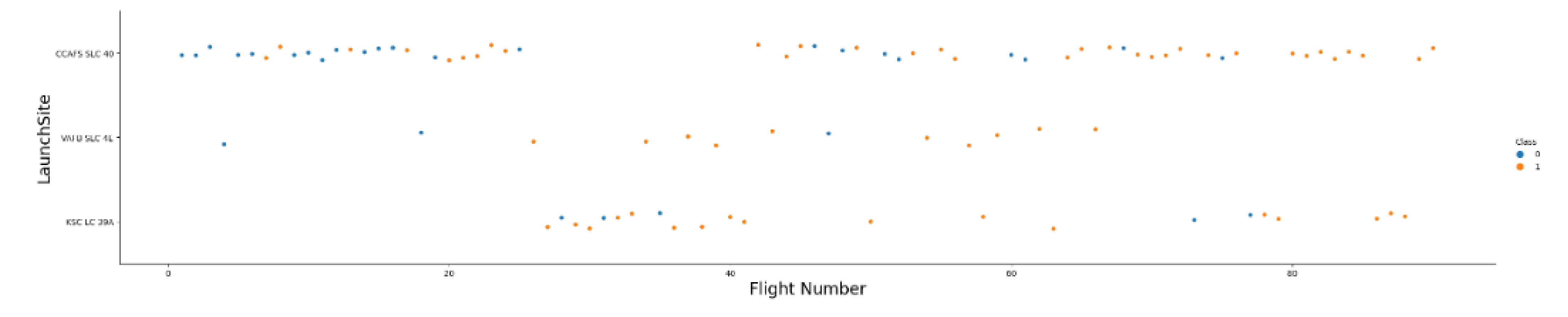
- Exploratory data analysis results
- Interactive analytics demo in screenshots
- Predictive analysis results

Section 2

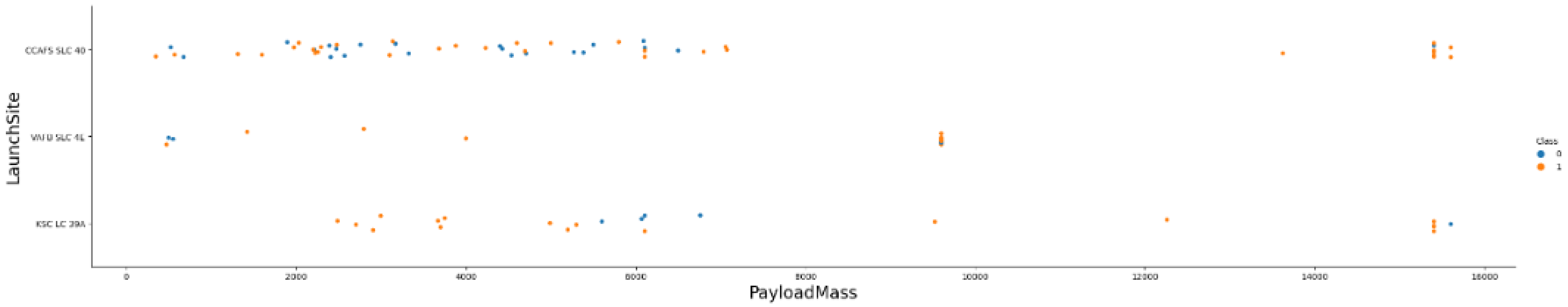
Insights drawn from EDA

Flight Number vs. Launch Site

- CCAF5 SLC 40 has most of recent launches were successful; while it's also possible that the LC39A is running as parallel or backups.



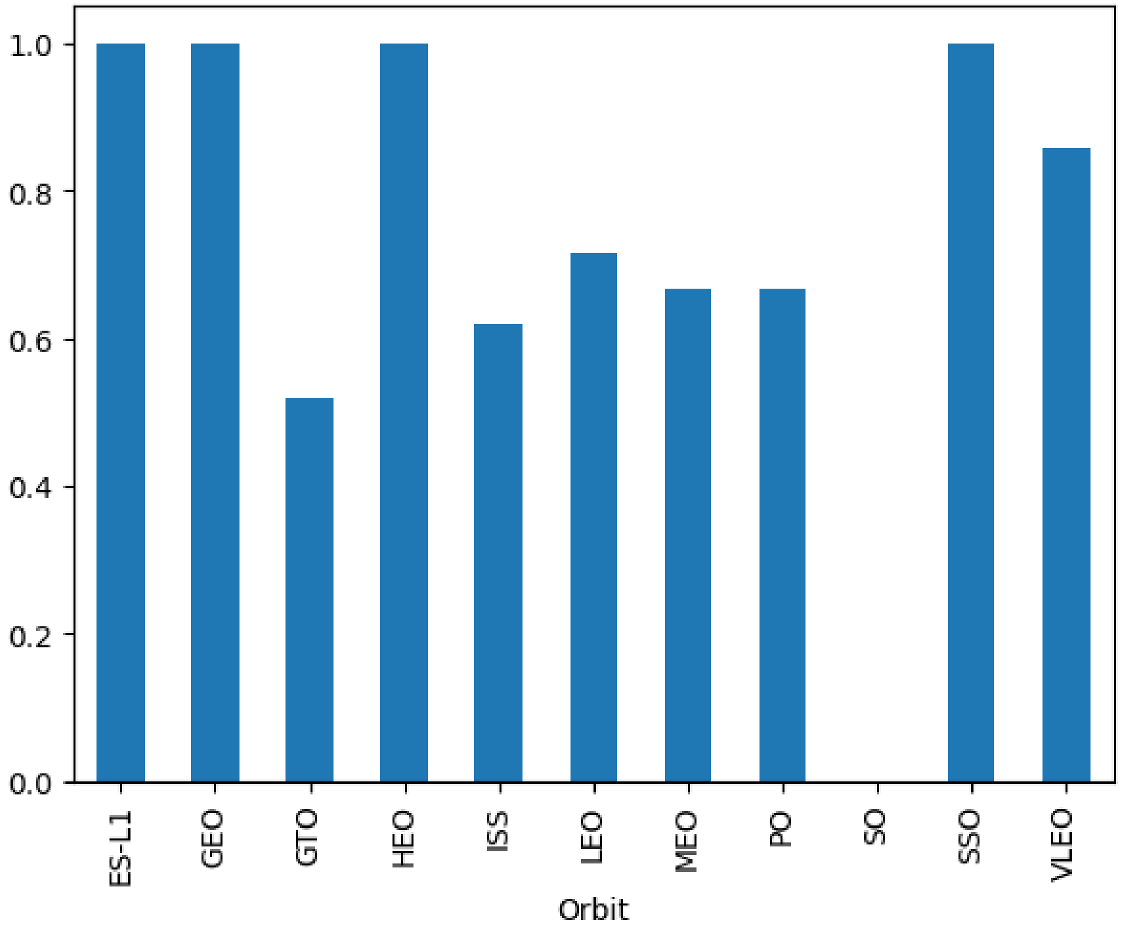
Payload vs. Launch Site



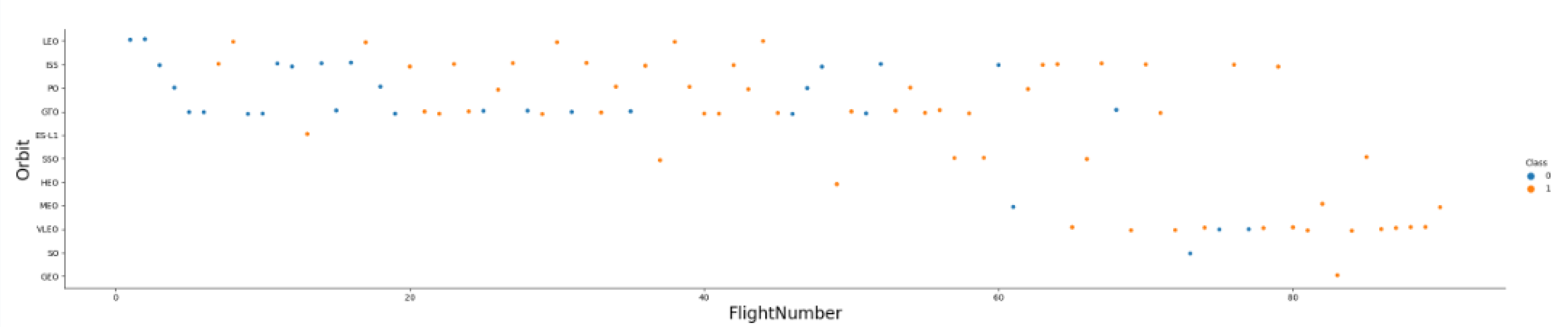
- Most Successful launches are below 7000kg.
- SLC 40 and LC 39A also had done some very high payload launches > 1500kg.

Success Rate vs. Orbit Type

- In this plot, ES-L1, GEO, HEO, SSO, VLEO had the most success rate.
- Others are all in the lower success rate group.



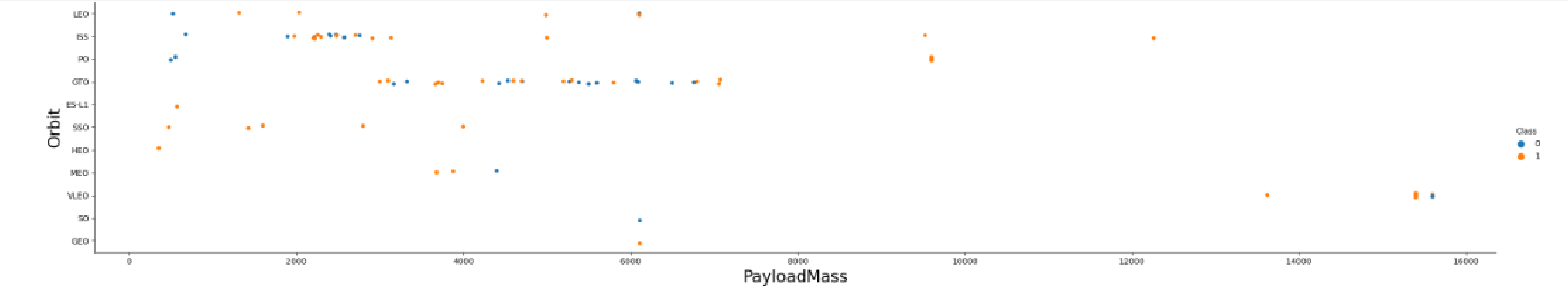
Flight Number vs. Orbit Type



- Later launches had better success rate.
- After #60 launch, most success orbit was VLEO.

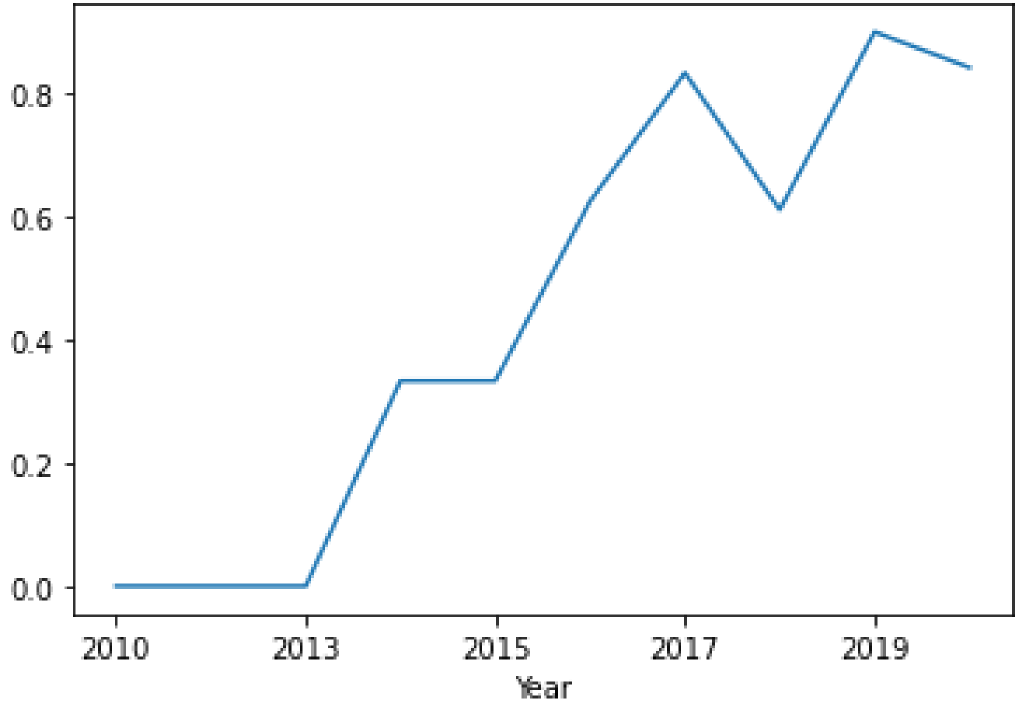
Payload vs. Orbit Type

- ISS and GTO done most launches in low and mid payload, ISS and VELO had some very heavy launches and were successful.



Launch Success Yearly Trend

- Great success increase from 2013 to 2017.
- Overall showing good success trend



All Launch Site Names

- Select **DISTINCT** to summarize unique launch site.

launch_site
CCAFS LC-40
CCAFS SLC-40
KSC LC-39A
VAFB SLC-4E

Display the names of the unique launch sites in the space mission

```
task_1 = '''
        SELECT DISTINCT LaunchSite
        FROM SpaceX
'''
create_pandas_df(task_1, database=conn)
```

	launchsite
0	KSC LC-39A
1	CCAFS LC-40
2	CCAFS SLC-40
3	VAFB SLC-4E

Launch Site Names Begin with 'CCA'

DATE	time__utc_	booster_version	launch_site	payload	payload_mass__kg_	orbit	customer	mission_outcome	landing__outcome
2010-06-04	18:45:00	F9 v1.0 B0003	CCAFS LC-40	Dragon Spacecraft Qualification Unit	0	LEO	SpaceX	Success	Failure (parachute)
2010-12-08	15:43:00	F9 v1.0 B0004	CCAFS LC-40	Dragon demo flight C1, two CubeSats, barrel of Brouere cheese	0	LEO (ISS)	NASA (COTS) NRO	Success	Failure (parachute)
2012-05-22	07:44:00	F9 v1.0 B0005	CCAFS LC-40	Dragon demo flight C2	525	LEO (ISS)	NASA (COTS)	Success	No attempt
2012-10-08	00:35:00	F9 v1.0 B0006	CCAFS LC-40	SpaceX CRS-1	500	LEO (ISS)	NASA (CRS)	Success	No attempt
2013-03-01	15:10:00	F9 v1.0 B0007	CCAFS LC-40	SpaceX CRS-2	677	LEO (ISS)	NASA (CRS)	Success	No attempt

- We used the keyword query and limit to 5 records

Total Payload Mass

- Used SUM function and filtered by NASA CRS.

sql SELECT SUM(PAYLOAD_MASS__KG_) AS TOTAL_PAYLOAD FROM SPACEXTBL WHERE PAYLOAD LIKE '%CRS%';

Total Payload = 111268

Average Payload Mass by F9 v1.1

- Calculate the average payload mass carried by booster version F9 v1.1
- Present your query result with a short explanation here
- SELECT AVG(PAYLOAD_MASS__KG_) AS AVG_PAYLOAD FROM SPACEXTBL WHERE BOOSTER_VERSION = 'F9 v1.1';

avg_payload
2928

First Successful Ground Landing Date

- SELECT MIN(DATE) AS FIRST_SUCCESS_GP FROM SPACEXTBL
- WHERE LANDING__OUTCOME = 'Success (ground pad)';
- First Successful landing: 2015-12-22

Successful Drone Ship Landing with Payload between 4000 and 6000

- SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL
- WHERE PAYLOAD_MASS__KG_ BETWEEN 4000 AND 6000 AND LANDING__OUTCOME = 'Success (drone ship)’;

booster_version
F9 FT B1021.2
F9 FT B1031.2
F9 FT B1022
F9 FT B1026

Total Number of Successful and Failure Mission Outcomes

- SELECT MISSION_OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL
- GROUP BY MISSION_OUTCOME ORDER BY MISSION_OUTCOME;

mission_outcome	qty
Failure (in flight)	1
Success	99
Success (payload status unclear)	1

Boosters Carried Maximum Payload

```
SELECT DISTINCT BOOSTER_VERSION FROM SPACEXTBL
WHERE PAYLOAD_MASS__KG_ = (SELECT MAX(PAYLOAD_MASS__KG_)
FROM SPACEXTBL)
ORDER BY BOOSTER_VERSION;
```

booster_version
F9 B5 B1048.4
F9 B5 B1048.5
F9 B5 B1049.4
F9 B5 B1049.5
F9 B5 B1049.7
F9 B5 B1051.3
F9 B5 B1051.4
F9 B5 B1051.6
F9 B5 B1056.4
F9 B5 B1058.3
F9 B5 B1060.2
F9 B5 B1060.3

2015 Launch Records

```
SELECT BOOSTER_VERSION, LAUNCH_SITE FROM SPACEXTBL
        WHERE LANDING__OUTCOME = 'Failure (drone ship)' AND DATE_PART('YEAR', DATE) = 2015
```

booster_version	launch_site
F9 v1.1 B1012	CCAFS LC-40
F9 v1.1 B1015	CCAFS LC-40

Rank Landing Outcomes Between 2010-06-04 and 2017-03-20

```
SELECT LANDING__OUTCOME, COUNT(*) AS QTY FROM SPACEXTBL
        WHERE DATE BETWEEN '2010-06-04' AND '2017-03-20'
        GROUP BY LANDING__OUTCOME ORDER BY QTY DESC;
```

landing__outcome	qty
No attempt	10
Failure (drone ship)	5
Success (drone ship)	5
Controlled (ocean)	3
Success (ground pad)	3
Failure (parachute)	2
Uncontrolled (ocean)	2
Precluded (drone ship)	1

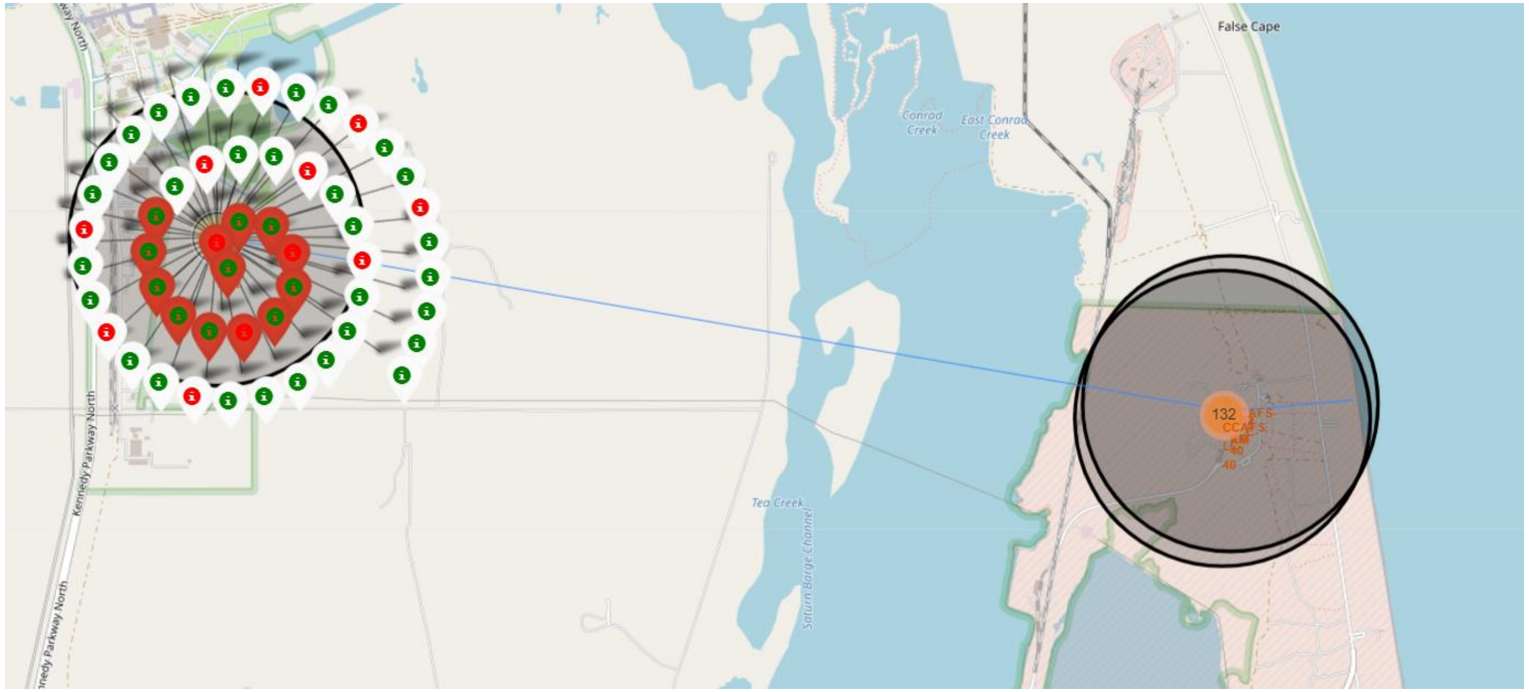
Section 4

Launch Sites Proximities Analysis

All launch sites overview

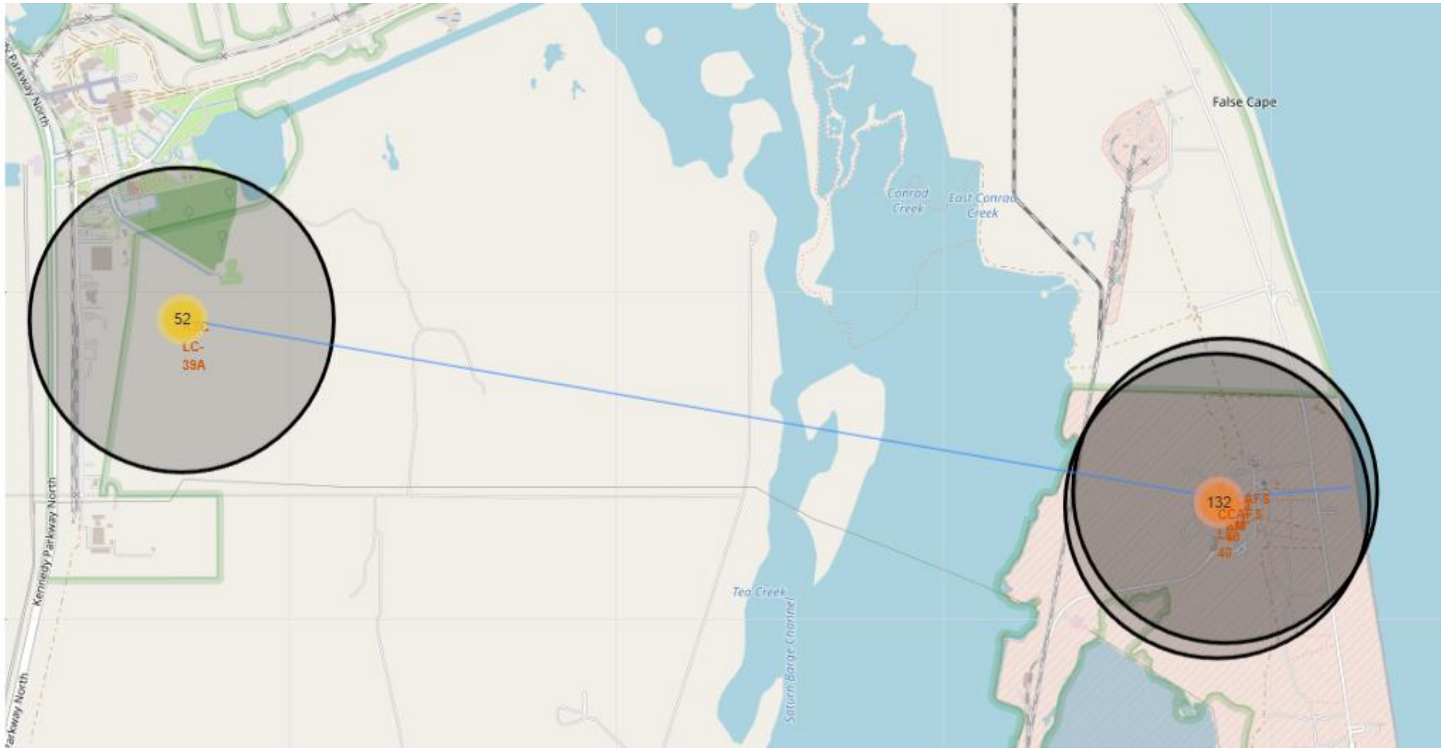
- Launch sites are near the coast, and south end area of the US.

Launch result of each site



- Showing successful and failed launches at each site

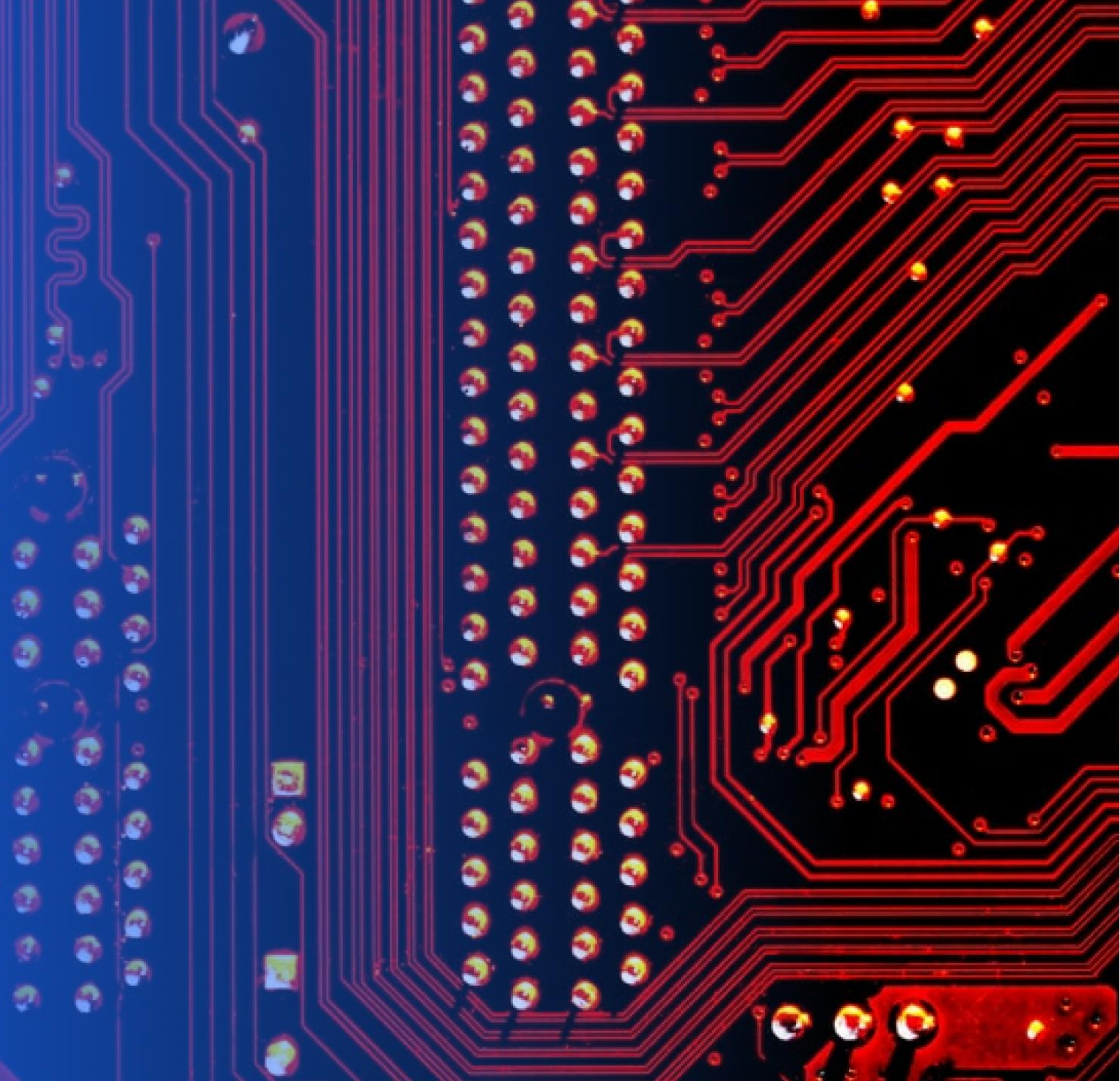
Launch location and safety consideration



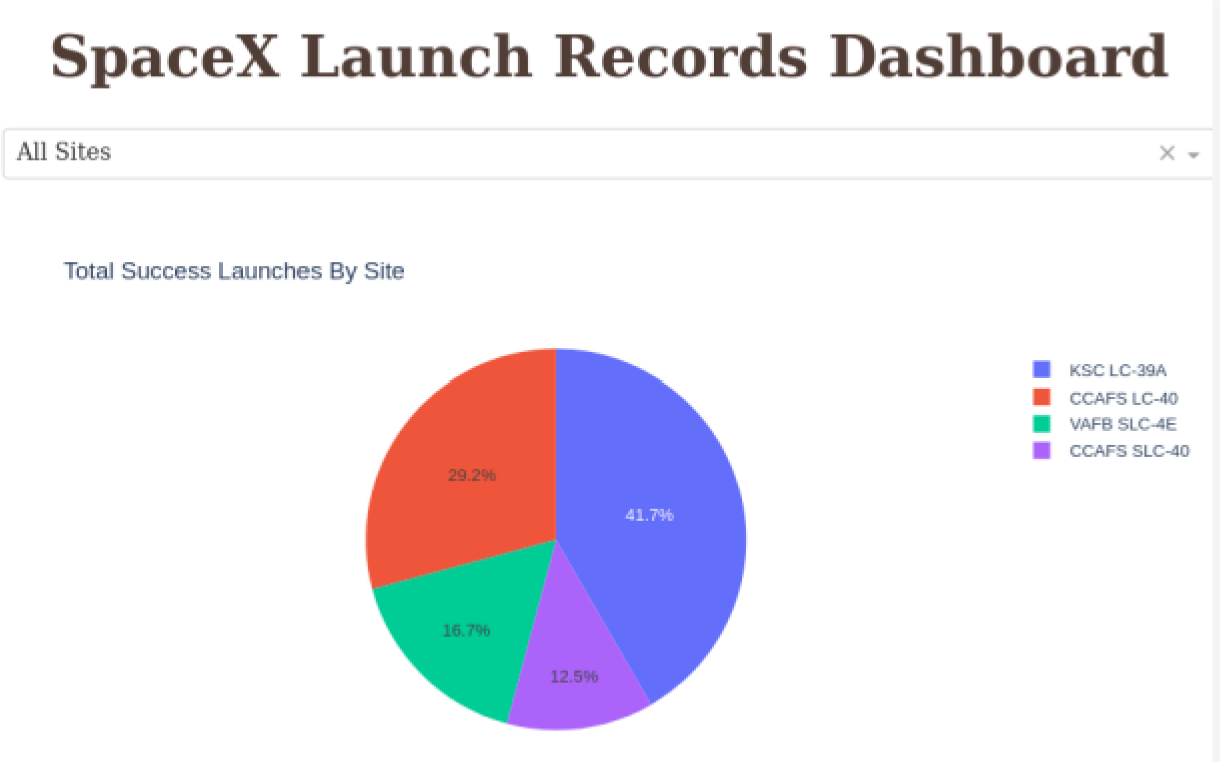
- Launch site SLC 40 is very close to the coast, and in distance from LC 39A

Section 5

Build a Dashboard with Plotly Dash

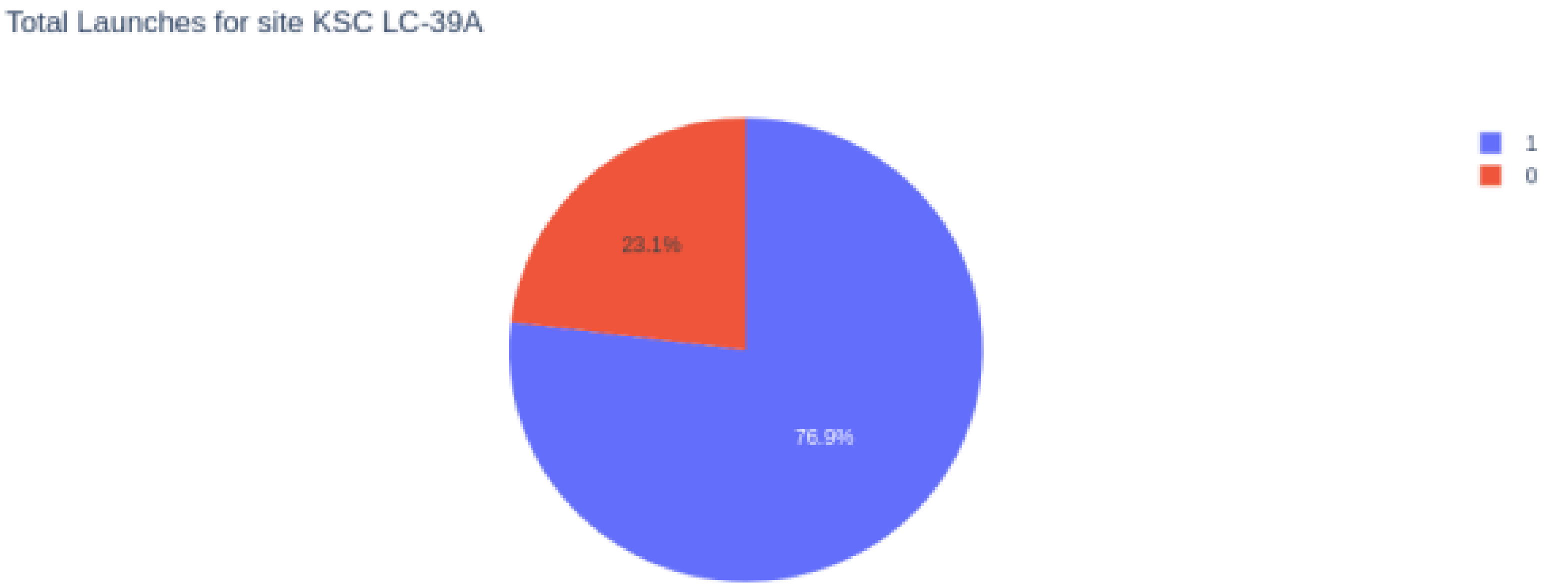


Pie Chart: Total Success Launches by Site



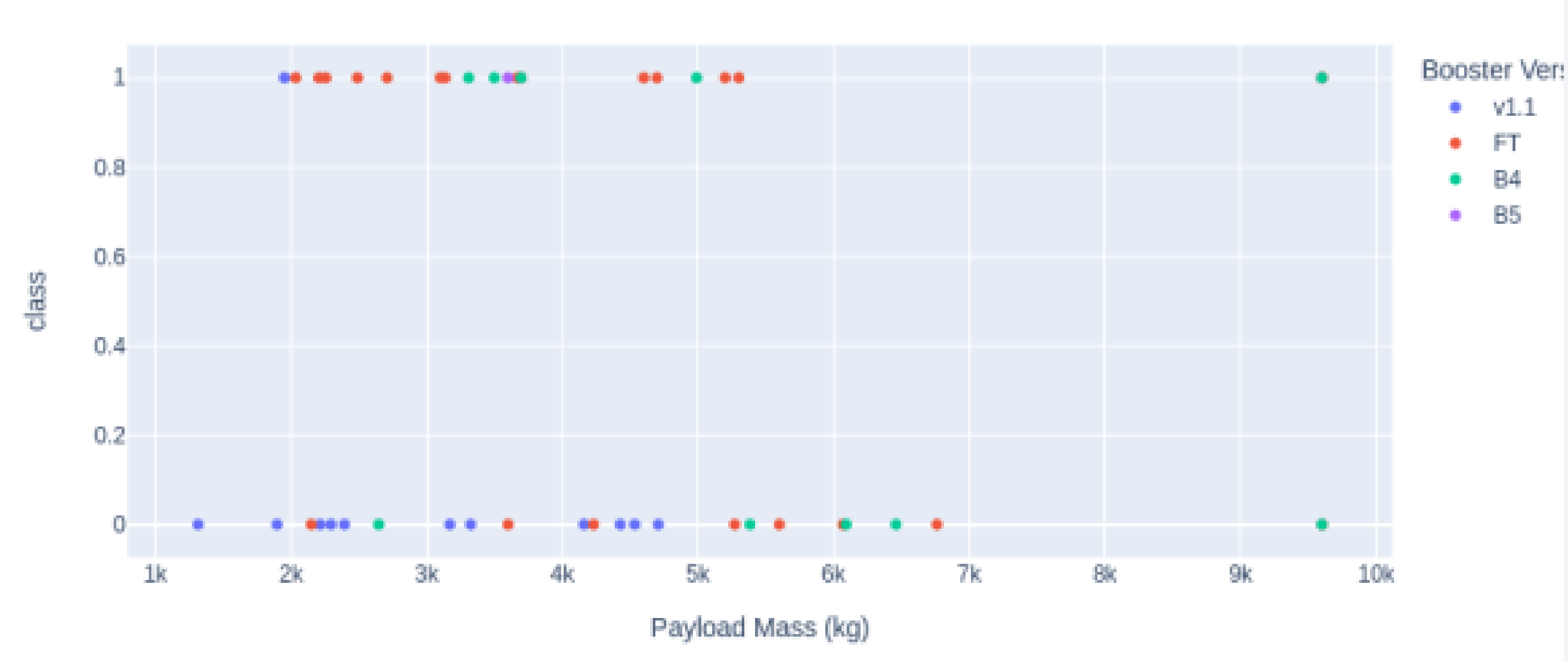
- Successful launches rely on the LC-39A site

Success launch ratio at the most successful site



- At LC 39A, 76.9% of launches are successful

Scatter plot: of Payload vs Launch Result, differentiated by Booster



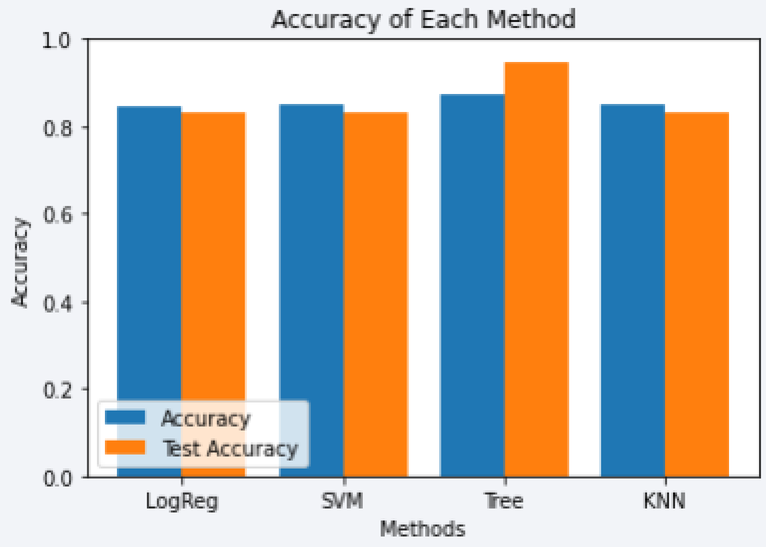
FT Booster with Payloads under 6000kg had better launch successes

Section 6

Predictive Analysis (Classification)

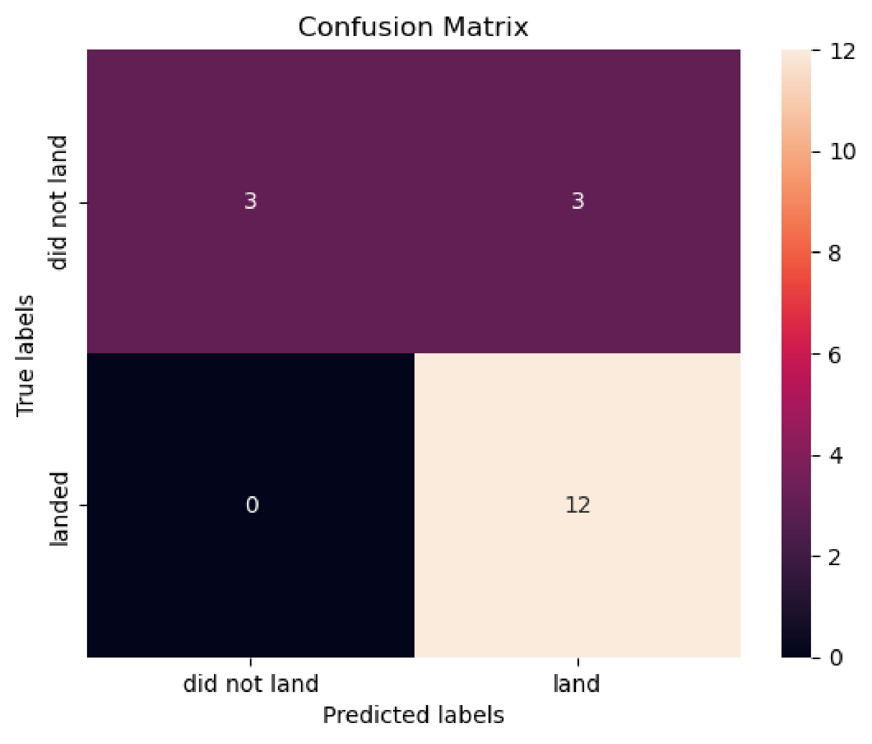
Classification Accuracy

- Tested Logistic Regression, SVM, Decision Tree and K -nearest
- Decision tree had the best test accuracy



Confusion Matrix of Decision Tree

- Confusion matrix showed high accuracy on high scores on predicted positive result and tested positive result.



Conclusions

- KSC LC-39A is the best launch site
- Orbits ES-L1, GEO, HEO, SSO, VLEO were relatively more successful than others.
- Launches above 6000kg are more successful, but high cost so less frequent.
- Decision tree for this case is more accurate.
- Launch site with more launches are easier to success.

Thank you!